IN THE CLAIMS

1. (Currently Amended) A database management system, comprising:

a processor configured to:

associate multiple different activities with a same transaction, each of the multiple different activities each consisting of a separate different associated subgroup of program instructions for the same transaction, for each of the different subgroups of program instructions, initiate a different associated subgroup of multiple different read and/or write actions that access an associated group of multiple different data items;

use and assign, for each of the different activities, only one single separate lock duration for all of the multiple different data items associated with the different associated subgroup of program instructions;

maintain, for each of the different activities, multiple different locks on all of the multiple different data items associated with the activity, and then release all of the multiple different locks for all of the multiple different data items associated with the activity together only when all of the subgroups of program instructions associated with the activity are completed so that all of the multiple different locks on all of the multiple different data items associated with the activity have a same lock duration; and

release all of the locks on a first one of the groups of the multiple different data items associated with a first one of the different activities of the same transaction while a second set of data items that include at least some of the first group of data items from the first activity, but that are associated with a second one of the different activities for the same transaction, remain locked for a second one of the separate lock durations associated with the second activity.

2. (Currently Amended) ~~A~~ The database management system according to claim 1 wherein one of the activities includes a group of individual shared lock operations and the processor activates locks for each of the individual shared lock

operations in the group and releases the locks only when ~~the~~ all of the individual shared lock operations in the group are completed.

3. (Currently Amended) ~~A~~ The database management system according to claim 1 including a memory containing an activity bit map that tracks individual activities for the transaction, the processor assigning activity identifiers to the activities according to the activity bit map.

4. (Currently Amended) ~~A~~ The database management system according to claim 1 wherein, for each of the different activities, the processor releases all of the multiple different locks associated with the activity in one operation only when the subgroup of program instructions associated with the activity is completed.

5. (Currently Amended) A method for locking data items in a database management system, comprising:

associating multiple different activities with a same transaction, each of the multiple different activities each consisting of a different associated subgroup of program instructions for the same transaction and each different subgroup of program instructions initiating a different associated subgroup of multiple different read and/or write actions accessing an associated group of multiple different data items;

associating and using, for each of the different activities, only a single separate individual lock duration for each different subgroup of program instructions and the accessed multiple different data items associated with the same activities in the same transaction;

maintaining, for each of the different activities, multiple different locks on all of the multiple different data items for each different subgroup of program instructions associated with the same activities; ~~and~~

releasing all of the multiple different locks for all of the multiple different data items associated with the same activities together only when the entire subgroup of program instructions associated with the same activities are all completed so that all of

the multiple different locks on all of the multiple different data items associated with the activity have a same lock duration; and

releasing all of the locks on a first one of the groups of the multiple different data items associated with a first one of the different activities of the same transaction while a second set of data items that include at least some of the first group of the data items from the first activity, but that are associated with a second one of the different activities for the same transaction, remain locked for a second one of the separate lock durations associated with the second activity.

6. (Currently Amended) ~~A~~ The method according to claim 5 further comprising releasing all of the locks on ~~a~~ the first one of the groups of multiple different data items associated with ~~a~~ the first one of the different activities of the same transaction while ~~a~~ the second set of data items that include at least some of the first group of the data items from the first activity, but that are ~~also~~ associated with a second one of the activities for the same transaction, remain locked for the separate individual lock duration associated with the second activity.

7. (Currently Amended) ~~A~~ The method according to claim 5 including:

maintaining an activity bit map that tracks individual activities for the transaction; and

assigning activity identifiers to the activities according to the activity bit map.

8. (Cancelled)

9. (Previously Presented) The method according to claim 5 including:

assigning a same unique activity identifier to multiple different arbitrary database access instructions that constitute one of the different activities in the transaction, the database access instructions performing one or more operations on multiple different data items in a database and the activity identifier assigned to and associated with the database

access instructions independently of any relationship that may exist between the multiple different data items in the database accessed by the database access instructions;

assigning multiple locks to the multiple data items corresponding with the operations performed on the multiple data items pursuant to the database access instructions; and

preventing other transactions and other associated activities from accessing the multiple different data items until all of the operations are completed for all of the database access instructions assigned to the activity identifier.

10.-13. (Cancelled)

14. (Currently Amended) Computer readable media containing instructions that when executed by a computer, comprise:

assigning activity identifiers to different individual subgroups of database access instructions within a same transaction that perform multiple different operations on multiple different data items in a database, the activity identifiers assignable to the database access instructions independently of any relationship that may exist between the multiple data items in the database accessed by the database access instructions;

assigning multiple locks to the multiple different data items corresponding with the operations performed on the multiple data items by the different subgroups of database access instructions;

assigning only one single separate lock duration for all of the multiple different data items associated with a same subgroup of database access instructions;

for each of the different subgroups of database access instructions, assigning ~~the~~ a same one of the activity identifiers to the locks that are associated with the subgroup of database access instructions;

releasing, for each of the assigned activity identifiers, all of the multiple locks on all of the multiple different data items assigned to the ~~assigned~~ activity identifier at the same time when all of the multiple operations are completed for the subgroup of database access instructions assigned to the ~~assigned~~ activity identifier, and wherein the releasing

of all of the multiple locks is in response to a single request via the assigned activity identifier; and[[.]]

releasing all of the locks on a first one of the subgroups of the multiple different data items associated with a first one of the activity identifiers of the same transaction while a second set of data items that include at least some of the first group of data items associated with the first one of the activity identifiers, but that are associated with a second one of the activity identifiers for the same transaction, remain locked for a second separate lock duration associated with the second activity identifier.

15. (Previously Presented) The computer readable media according to claim 14 including instructions that when executed assign the activity identifiers to an arbitrary group of related database access instructions performing operations on an arbitrarily related group of data items.

16. (Previously Presented) The computer readable media according to claim 14 including instructions that when executed assign common transaction identifiers to different related groups of database access instructions each assigned different activity identifiers and coordinate when the related groups of database access instructions are allowed to perform operations on the data items.

17. (Previously Presented) The computer readable media according to claim 14 including instructions that when executed assign a first transaction identifier to a group of individual shared operations, assign locks to the data items associated with the shared operations, and hold the locks until all of the individual shared operations in the group have been completed.

18.-24. (Cancelled)